

## **REMARKS**

### **Claim objections**

Applicant gratefully acknowledges the helpful suggestions made by the Examiner and has followed these in the amended claims above, with the exception of the following particular points, for which further consideration is respectfully requested:

Claim 3, line 2: As this is a dependency issue, reference is made to claim 1 which recited "at least one other object", while claim 3 recited "the step of modifying said at least one other object". It is respectfully submitted that "said at least one other object" is the correct form of wording rather than "said at least said one other object".

Claim 3, line 2: The Office Action suggested replacing "a reference" with "said reference". The same point is also raised in the rejection of this claim under 35 USC 112, where it is questioned whether the invention involves one reference or multiple references.

It is respectfully submitted that there are multiple references involved in the invention of claim 3, and that there is no lack of clarity in this regard. Claim 1 distinctly recites the step of adding to the new software object (i.e. that object whose creation is explicitly recited in the second step of the claim), a reference to the at least one other software object also mentioned in claim 1. Claim 3, on the other hand, distinctly recites modifying that "said at least one other software object" with a reference back to the new software object.

In other words, claim 1 recites a reference in one direction from the new object to an existing object, and claim 3 recites a reverse reference from the existing object to the new object. These references between software objects clearly exist distinctly from one another. Accordingly, it would be inappropriate to change "a reference" to "said reference" in claim 3. The question of what the word "reference" actually means in the context of this invention will be dealt with in response to the 35 USC § 112 rejection below.

Claim 4, line 3: Applicants suggest that it would be inappropriate to add "the" or "said" before the word "objects" in this claim. Claim 1 mentions certain software

objects which relate to contacts having a shared or similar skillset identifier and are therefore in a single queue. Claim 4, on the other hand, covers the situation where a contact is assigned multiple skillset identifiers and whose software object therefore has separate references added to software objects in different queues. The software objects in different queues have no antecedent basis to the software objects previously recited in claim 1.

Accordingly, it would be inappropriate to identify the software objects in different queues, which are recited in claim 4, with the words “the” or “said”.

Claim 7, line 4: This claim has been fundamentally amended and a new claim 17 added, rendering the objection moot.

Claim 11: This is an independent claim and the recitation of “a received contact” in this claim is thus a new feature with no antecedent basis. Accordingly, it would be inappropriate to suggest an antecedent basis with the words “the” or “said”.

Claim 11, line 5: The same comments apply as in the preceding paragraph, in that this is the first recitation in an independent claim of “at least one other software object”.

Claim 12, line 10: Rather than the words “determine which said object”, the claim has been amended to read “determine which of said software objects”, as it is respectfully submitted that this wording is clearer.

Claim 13: For the reasons given above in relation to claim 11, there is no antecedent basis for “reference” in this independent claim, and thus the words “the” or “said” are not added as suggested.

#### Claim rejections 35 USC § 101

Claim 1 has been amended to recite the step of storing the collection of software objects in the memory and it is submitted that this produces a “real world” or tangible output, thereby overcoming the rejection to claims 1 – 7.

Regarding claims 11 and 12, the computer program products, and regarding claims 15 and 16, the software object and the virtual queue of contacts, respectively, are all now recited as being “embodied on a computer-readable medium” thereby

rendering each claim as fulfilling the requirement of functionally descriptive material embodied on a computer-readable medium which is therefore statutory.

Claim rejections 35 USC § 112

Throughout the claims, the software object created in e.g. the second step of claim 1, is referred to consistently as the “new software object”. Throughout the claims, the various objects are now consistently referred to as “software objects”, to overcome the objection raised at multiple points that the term objects might relate to some other kind of object.

It is submitted that the amendments made in line with the Examiner's suggestions in the “Objections” part of the Office Action have the effect of clarifying a number of the Section 112 rejections and, therefore, the comments below will be confined to those requiring separate commentary.

Claim 3, line 2: Applicants addressed previously the question of whether there is one reference or multiple references. However, as regards the allegation that the word “reference” is itself confusing and unclear in relation to contacts within a call centre, it is respectfully pointed out that the word “reference” is not used at all in relation to contacts within a call centre. The “reference” of the claims is a reference found in the software objects representing such contacts.

Software objects which contain references to other software objects are well known to any programmer. While not limited to object-oriented programming, this is the predominant type of software object with which the skilled person will be most familiar. One of the fundamental qualities that such a software object may have is a reference to (also known as a “pointer” to) another software object. Such “references” are entirely clear to the skilled person who would be a programmer responsible for the systems which would create the software objects of the present invention.

It is respectfully suggested that the uncertainty expressed in the Office Action may derive from the assumption that the contacts themselves contained references to other contacts when in fact it is the software objects which are created and which contain references to other objects allowing a collection of such inter-referenced

objects to act as a queue. This understanding is crucial to a proper appreciation of the patentability of the invention as well.

Claim 5, line 3: The phrase “the head of a queue” has been amended to “a head of a queue”. It is respectfully submitted that this overcomes the rejection. The skilled person will know how to identify the head of a queue.

Claim 6, line 4: The amended claims refers to “the new software object”, rather than the term which gave rise to this rejection. This phrase clearly refers to the “new software object” created in claim 1.

Claim 7: This claim originally dealt with two alternatives, which were described in the first complete paragraph of page 12 of the application as filed. For increased clarity, claim 7 now has been amended to refer only with the first of these options, i.e. that in which the contact manager memory space and the queuing module memory space form part of a common space.

The second of these options, involving a replication service, has been claimed in new claim 17. This claim makes it clear that there are two memory spaces, namely a contact manager memory space and a queuing module memory space, each of which stores a copy of each of the software objects recited in claim 6. A replication service is provided which ensures that changes in the first of the copies are reflected in corresponding changes to the second of the copies and *vice versa*. It is submitted that this clarifies the wording on which the rejection was raised.

#### Claim rejections 35 USC § 102

##### **Independent claim 1**

The disclosure of Mears et al. fails to disclose certain limitations of amended claim 1, and in particular Applicants would request that the Examiner consider the following points:

##### 1. “Creating a new software object for said received contact”

This new software object is alleged to be found in the creation of a “contact object” at col. 42, lines 10-12, and indeed the Office Action explicitly equates the term “software object” with “contact object”.

Applicants submit that the “contact object” of Mears et al. is not in fact a software object at all. The contact object of Mears fulfils the function of recording data about the contact which can be used to route the call to an agent. A “software object” according to claim 1, however, is a more sophisticated entity which must be able to include references to other software objects, as will be discussed below in more detail.

2. “Determining a queuing position for said new software object relative to at least one other software object representing a contact having a skillset identifier similar to said skillset identifier assigned to said received contact”

The passage relied on (col. 47, lines 5-25) discloses the prioritization of contacts, not of software objects. Furthermore, it does not mention a queuing determination as between objects representing contacts having similar skillset identifiers: all of the contacts in Mears appear to be queued to a common queue and not to be queued according to skillset.

3. “Adding to said new software object a reference to said at least one other software object”

This is at the heart of why the contact objects of Mears cannot be regarded as software objects (see point 1 above). The Office Action states that this feature is disclosed at col. 46, lines 31-40 of Mears. That passage contains no such disclosure and is accurately summarized in the Office Action as follows:

*“the email proxy proceeds with routing the contact by instructing the routing manager to begin the process of adding contacts to the queue.”*

Applicants wish to point out that the claim limitation under discussion involves the modification of a software object representing a contact, and in particular, the addition to such a software object of a reference to another software object. Mears provides no such teaching whatsoever.

4. “Storing in memory a collection of said software objects each containing said reference to at least one other software object; whereby said stored collection of said software objects provides a prioritised queue”

Mears discloses, at the most, the storage in memory of a collection of contact objects each of which has a place in a queue which is maintained separately by the system. This misses the crucial benefit of this invention – namely that the invention allows one to inherently provide for a prioritised queue simply by maintaining a collection of software objects each of which points to another of the software objects.

In other words, the collection of software objects in the present invention effectively is the queue; while for Mears the collection of “contact objects” requires the maintenance of a separate queue to have any usefulness. The feature underlying this fundamental difference, which has been referred to several times above, is the priority-based reference or pointer from the newly created software object to another existing software object. The net effect of adding references from each new software object to existing software objects is that the aggregate collection of such objects has the queuing information built into it.

5. “...a prioritised queue for a skillset”

The closing phrase of claim 1 are highlighted here because it is important to note that the invention of claim 1 relates to a method which ultimately provides a queue for a skillset. Mears does not suggest a queue for a particular skillset; rather it suggests that one of the attributes of a particular contact object is the skills required to service it. At no point is it suggested that different, skillset-specific queues can be provided.

**Dependent claims 2-6**

Claims 2-6, being dependent on claim 1, benefit from the patentability of claim 1 for at least the same reasons as are given above.

**Independent claim 8**

Regarding claim 8, Applicants fail to find any suggestion in the rejection made in the Office Action, of where Mears might provide a teaching of any of the following features:

- a network of contact centres;

- each contact being represented by a software object maintained at one of said contact centres;

- each said software object containing references to one or more other of said software objects maintained at the same contact centre to provide a queue of software objects at each said contact centre;

While these features are to be found in the preamble, they cannot be ignored for the assessment of patentability since each case must be evaluated on its merits according to MPEP 2111.02, which goes on to quote:

"[A] claim preamble has the import that the claim as a whole suggests for it." *Bell Communications Research, Inc. v. Vitalink Communications Corp.*, 55 F.3d 615, 620, 34 USPQ2d 1816, 1820 (Fed. Cir. 1995). "If the claim preamble, when read in the context of the entire claim, recites limitations of the claim, or, if the claim preamble is 'necessary to give life, meaning, and vitality' to the claim, then the claim preamble should be construed as if in the balance of the claim." *Pitney Bowes, Inc. v. Hewlett-Packard Co.*, 182 F.3d 1298, 1305, 51 USPQ2d 1161, 1165-66 (Fed. Cir. 1999)

In the present case, the claim depends entirely for its meaning on the preamble. This can be seen from the fact that the application of the Mears reference against claim 8 fails to take any account of the fact that the method of claim 8 only works in a network of contact centers, and Mears discloses no such network. For this same reason, the Office Action simply seems to ignore all of the network-related aspects of the claim which are discussed.

Claim 8 further requires the following features:

"requesting from each said contact centre the highest priority queued software object matching said criteria"

It is noted that this feature is repeated in the rejection but Mears is not applied to it, i.e. no guidance is provided where Mears might disclose such a request issued to a plurality of contact centers.

"receiving information relating to each such highest priority queued software object from said contact centres"

The Office Action argues that this clause is met by the retrieval of the top item in a single queue as taught at col. 47, lines 28-38 of Mears. In fact, the clause requires that a plurality of software objects are identified, each belonging to a different contact center.

Accordingly, the features of claim 8 are neither taught nor suggested by Mears et al.

#### **Dependent claim 9**

Claim 9, being dependent on claim 8, benefits from the patentability of claim 8 for at least the same reasons as are given above.

#### **Claim 11**

The arguments made above in relation to claim 1 apply *mutatis mutandis* to claim 11.

#### **Claim 12**

The arguments made above in relation to claim 8 apply *mutatis mutandis* to claim 12.

#### **Claim 13**

The arguments made above in relation to claim 1 apply *mutatis mutandis* to claim 13.

#### **Claim 14**

The arguments made above in relation to claim 8 apply *mutatis mutandis* to claim 14.

#### **Claim 15**

The arguments made above in relation to claim 1 apply *mutatis mutandis* to claim 15 insofar as they discuss the failure of Mears to disclose any software objects having references to other software objects.



### **Claim 16**

The arguments made above in relation to claim 1 apply *mutatis mutandis* to claim 16 insofar as they discuss the failure of Mears to disclose any software objects having references to other software objects or a queue formed of such software objects.

### Claim rejections 35 USC § 103

### **Claim 7**

The rejection of claim 7 implicitly relies on the rejection of claim 6 from which it depends and accordingly, the arguments provided above apply equally to claim 7.

However, the further teaching of Wallenius et al. fails to even supply the missing limitations of claim 7 which the Office Action acknowledges to be missing from Mears.

After noting that Mears fails to disclose a memory space in the contact manager and a memory space in the queuing module, each of which stores software objects, Wallenius is relied on for its disclosure of how a database may be implemented in memory. This does not touch on the requirement that both the contact manager and the queuing module must each provide a memory space.

Without some teaching of this dual memory space (or shared memory space) there is no incentive to combine Wallenius and Mears other than the suggestion to found in hindsight with knowledge of the invention. It is stressed that this argument is made without prejudice to the primary argument that Mears fails to anticipate the base claim on which claim 7 depends.

For completeness it is pointed out the the subject-matter of new claim 17, which is based on part of original claim 7, is not suggested in any way by either Mears or Wallenius.

### **Claim 10**

The arguments made above in relation to the preamble of claim 8 applies *mutatis mutandis* to claim 10.

Furthermore, the disclosure of Lambert does not have the relevance attributed to it. Lambert does not, as alleged, disclose the copying of contact objects in a queue. Lambert discloses that a queue object, which is a software object representing the entire queue, may be retrieved, copied, the copy updated, and that updated copy sent back to its source where it may replace the original. This is not relevant to the mimicking of changes in a software object relating to a contact held at one contact center by making changes in a copy which is held at another contact center.

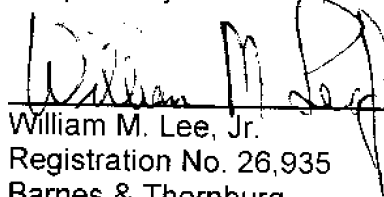
For this reason, and without prejudice to the argument made in relation to the apparent lack of patentable weight accorded to the preamble features, the suggested combination of Mears and Lambert does not result in the claimed invention of claim 10.

### **Summary**

In view of the amendments and arguments made herein, the applicants respectfully request the examiner withdraw the rejections, and allow the application.

August 7, 2007

Respectfully submitted



William M. Lee, Jr.  
Registration No. 26,935  
Barnes & Thornburg  
P.O. Box 2786  
Chicago, Illinois 60690-2786  
(312) 214-4800  
(312) 759-5646 (fax)